

CSCI 1120 (Low-Level Computing), Fall 2008

Homework 3

Assigned: October 8, 2008.

Due: October 13, 2008, at 5pm.

Credit: 20 points.

1 Reading

Be sure you have read, or at least skimmed, the readings for 10/06, linked from the “Lecture topics and assignments” page¹.

2 Programming Problems

Do the following programming problems. You will end up with at least one code file per problem. Submit your program source (and any other needed files) by sending mail to `bmassing@cs.trinity.edu`, with each file as an attachment. Please use a subject line that mentions the course number and the assignment (e.g., “csci 1120 homework 3”). You can develop your programs on any system that provides the needed functionality, but I will test them on one of the department’s Linux machines, so you should probably make sure they work in that environment before turning them in.

1. (10 points) Complete the sort program we wrote in class 10/06 by filling in the `sort` function. (You can find it linked from the course “sample programs” page here².)

It’s completely up to you which sorting algorithm to implement, though I’m inclined to recommend that you just do one of the simple-but-slow ones (e.g., bubble sort or selection sort). We will probably do at least one of the faster recursive ones (quicksort and mergesort) later in the semester. Please say in comments at the start of the program which sorting algorithm you’re implementing. Feel free to make other alterations to the program (e.g., adding more functions, though depending on your choice of algorithm you might want to just do everything in `sort`).

2. (10 points) Write a C program that reads text from standard input, up to end of file (control-D if reading from the terminal), and prints, for each possible character, how many times it occurs in the input. Print information only about the characters actually present in the file. For example, your program should print something like the following when input is the “hello, world” sample program:

```
enter some text, control-D to end
count for character 9 (not printable) is 2
count for character 10 (not printable) is 10
count for character 32 ( ) is 17
```

¹http://www.cs.trinity.edu/~bmassing/Classes/CS1120_2008fall/HTML/schedule.html

²http://www.cs.trinity.edu/~bmassing/Classes/CS1120_2008fall/SamplePrograms/Programs/sort.c

```
count for character 34 (") is 6
count for character 35 (#) is 1
count for character 40 (( is 2
count for character 41 ()) is 2
count for character 42 (*) is 3
count for character 44 (,) is 2
count for character 46 (.) is 2
count for character 47 (/) is 2
count for character 48 (0) is 1
count for character 58 (:) is 1
count for character 59 (;) is 2
count for character 60 (<) is 1
count for character 62 (>) is 1
count for character 72 (H) is 1
count for character 80 (P) is 1
count for character 92 (\) is 1
count for character 97 (a) is 4
count for character 99 (c) is 1
count for character 100 (d) is 8
count for character 101 (e) is 5
count for character 102 (f) is 1
count for character 103 (g) is 1
count for character 104 (h) is 3
count for character 105 (i) is 7
count for character 108 (l) is 10
count for character 109 (m) is 2
count for character 110 (n) is 8
count for character 111 (o) is 11
count for character 112 (p) is 3
count for character 114 (r) is 10
count for character 115 (s) is 3
count for character 116 (t) is 9
count for character 117 (u) is 4
count for character 118 (v) is 1
count for character 119 (w) is 3
count for character 123 ({) is 1
count for character 125 (}) is 1
```

(Remember that you can use input redirection to get input from a file rather than the keyboard.)

Hints:

- Characters are actually small integers, and can be treated as such. If `c` is an `int` representing a character, you can print its numeric value and its value as a character with a line such as the following:
`printf("%d %c\n", c, (char) c);`
- Possible integer values for characters range from 0 through `UCHAR_MAX`, where `UCHAR_MAX` is a constant defined in `limits.h`.

- Feel free to copy code from the “copy characters” program we wrote in class. (You can find it linked from the course “sample programs” page [here](#)³.)

³http://www.cs.trinity.edu/~bmassing/Classes/CS1120_2008fall/SamplePrograms/Programs/copy.c